



EOHHS CTO Organization
Secretariat Application Reference Manual

Version 1.0

EOHHS CTO Organization	Version: 1.0
Secretariat Application Reference Manual	Date: February 02, 2011

Revision History

Date	Version	Description	Author(s)
Feb 2, 2011	1.0	Initial Revision	

EOHHS CTO Organization	Version: 1.0
Secretariat Application Reference Manual	Date: February 02, 2011

Table of Contents

1.	INTRODUCTION.....	6
1.1	Purpose	6
1.2	Business Drivers	6
1.3	Information Technology Drivers	6
1.4	How this document is organized	7
2.	DESKTOP AND LAN SERVICES.....	8
2.1	Standard Operating Environment	8
2.2	Servers	8
2.3	Telephony	9
2.4	Assistive Technology	9
2.5	Security	9
2.6	Information Technology Acquisition Policy	9
2.7	Desktop Power Management Policy	10
2.8	Open Standards Policy	10
3.	WEBSITE INFORMATION ARCHITECTURE SERVICES.....	11
3.1	Access Channels.....	11
3.1.1	Portal	11
3.1.2	Virtual Gateway	11
3.1.3	XML Gateway	11
3.1.4	IVR	11
3.1.5	Mobile Devices	11
3.2	Web Site Policies & Guidance	11
3.3	Accessibility.....	11
4.	APPLICATION SERVICES	12

EOHHS CTO Organization	Version: 1.0
Secretariat Application Reference Manual	Date: February 02, 2011

4.1	Project Management.....	12
4.1.1	Methodology	12
4.1.2	Collaboration.....	12
4.2	Enterprise Services	12
4.2.1	Access and Identity Management Services (AIMS)	12
4.2.2	Enterprise Document Management (EDM)	12
4.2.3	XML Gateway	13
4.2.4	File Transfer Services	13
4.2.5	Enterprise Service Bus.....	13
4.2.6	Virus Scan Service.....	13
4.2.7	HHS Reporting Service.....	14
4.2.8	HHS Interchange	14
4.2.9	Log Services.....	14
4.2.10	Application Monitoring.....	14
4.2.11	Address Verification.....	14
4.2.12	Geographical Information Services (GIS).....	14
4.2.13	Quality Assurance Services.....	15
4.2.14	Release Management.....	15
4.2.15	Backup and Recovery.....	15
4.2.16	HL7 Gateway Service.....	15
4.2.17	Sybase Translator Service	15
4.2.18	WTX Service	15
4.3	Architecture	15
4.3.1	Architectural Principles	16
4.3.2	Quality of Service.....	16
4.3.3	Conceptual View.....	17
4.3.4	Logical View	18
4.3.5	Process View.....	27
4.3.6	Deployment View	29
4.3.7	Technology Guidelines	31
4.4	Requirements	39
4.5	Development.....	39
4.5.1	Integrated Development Environment.....	39
4.5.2	Project Structure.....	40
4.5.3	Build Tool	41
4.5.4	Continuous Integration.....	41
4.5.5	Code Quality	41
4.5.6	XML Editor	42
4.6	Testing	42
4.6.1	Unit Testing	43
4.6.2	Functional Testing	44
4.6.3	Regression Testing.....	44
4.6.4	Performance Testing.....	45
4.6.5	Accessibility Testing.....	46

EOHHS CTO Organization	Version: 1.0
Secretariat Application Reference Manual	Date: February 02, 2011

4.6.6	Vulnerability Testing.....	46
4.6.7	Defect Tracking	46
4.6.8	Performance Monitoring	47
4.7	Environment	48
4.7.1	Application Categories	48
4.7.2	Application Environments	49
4.7.3	Virtual Gateway.....	49
4.7.4	Virtual Datacenter	50
4.8	Release Management	50
4.9	Configuration Management	50
4.10	Change/Risk/Issue Management.....	50
5.	HELPDESK SERVICES	51

EOHHS CTO Organization	Version: 1.0
Secretariat Application Reference Manual	Date: February 02, 2011

1. Introduction

Commonwealth of Massachusetts (Information Technology) IT strategic plan identified key initiatives in an effort to modernize its aging systems and consolidate fragmented IT services. The initiatives mandate infrastructure based on Service Oriented Architecture (SOA) model and consolidation of IT services at 3 organization levels Commonwealth, Secretariat and Agency. This will allow the agencies to deliver the services more effectively. It promotes reuse, sharing of resources and services thus eliminating incompatibility within agencies. It will provide flexible infrastructure that is the foundation to fulfill the drivers listed in following sections.

1.1 Purpose

This document provides guidance on the standards, technologies, and tools at the secretariat level, to be used in the Health and Human Service (HHS) computing environment. It also provides guidelines for application architecture for ensuring that individual projects conform to the overall architectural vision for HHS and is based on a blueprint for smart, standard, open, service-oriented services. This document is intended to be used enterprise-wide within HHS by which new initiatives leverage the new architecture, process, and organization, ensuring that every initiative across the enterprise achieves consistently high quality.

Enterprise Technical Reference Manual (ETRM) published by Massachusetts Information Technology Division represents the standards that all organizations are expected to adhere to within the Commonwealth. This document acts as a super set of the ETRM, inheriting anything in it, and expanding and providing detail as necessary. If your business requirement cannot be met by technologies listed in this document, HHS Chief Technology Office should be consulted prior to architecture review.

1.2 Business Drivers

The Information Systems (IS) organization must provide non-IS management with the ability to:

- Access business information faster, cheaper and easier
- React to changing business and legislative environment
- Accommodate organizational restructuring
- Accommodate business growth and expansion
- Rapidly expose business services to interested parties
- Control the cost of business processes and IS development
- Eliminate reliance on inflexible and unstable existing systems and processes
- Accommodate existing information technology assets (legacy systems)
- Allow for cost-effective retirement of outdated systems
- Maintain quality and accuracy of information

1.3 Information Technology Drivers

In addition to always keeping the business drivers in focus, the following information technology drivers need to be addressed:

EOHHS CTO Organization	Version: 1.0
Secretariat Application Reference Manual	Date: February 02, 2011

- Consolidate development tools and technologies
- Improve reliability, modularity, extensibility, scalability and stability of existing and new systems
- Establish technical guidance by promoting best practices and standards
- Exploit new network computing architectures, particularly those based on Web Services standards so that future evolution of the architecture is supported
- Use packaged applications (COTS) where appropriate
- Replace obsolete and outdated technologies
- Provide integration mechanisms for non-conformant legacy systems
- Provide a repository of reusable software components that can be leveraged in projects within HHS
- Provide integration mechanism for existing system to promote the concept of service-oriented architecture
- Provide security requirements in order to protect the HHS infrastructure from unauthorized access
- Provide solutions that are based on open-standards and vendor independent implementations of those standards and specifications
- Use Open-Source Software where appropriate
- Establish a process for defining standards
- Adhere to Commonwealth IT Standards and Policies

1.4 How this document is organized

The IT consolidation framework identifies the following IT services to be consolidated at a secretariat level.

- Desktop and Local Area Network (LAN) Services
- Website Information Architecture Services
- Application Services
- Helpdesk Services

This document is divided into sections reflecting the above services.

Desktop and Local Area Network (LAN) Services – This section provides information about the standard operating environment.

Website Information Architecture Services – This section provides information about various channels for accessing information.

Application Services – This section provides a standard technical architecture for all EOHHS technical business and shared services. It also lists the different choices available to project development team when developing an application and still stay within the boundaries of the architecture.

Helpdesk Services – This section provides information about the support services.

EOHHS CTO Organization	Version: 1.0
Secretariat Application Reference Manual	Date: February 02, 2011

2. Desktop and LAN Services

2.1 Standard Operating Environment

Standard operating environment guidelines adopted by ITD defined must be followed. More information is available at:

<http://www.mass.gov/?pageID=afsubtopic&L=5&L0=Home&L1=Research+%26+Technology&L2=IT+Policies%2c+Standards+%26+Guidance&L3=Enterprise+Architecture&L4=Information+Technology+Architecture+and+Enterprise+Standards&sid=Eoaf>

NOTE: These standards are being updated. Until then the following are the standard for HHS operating environment.

Purpose	Standard	Version
Operating System	Windows XP	With Service Pack 2
Browser	Internet Explorer	6.x or higher
Word Processor	Microsoft Word	2003
Spread Sheet	Microsoft Excel	2003
Presentation	Microsoft PowerPoint	2003
Diagram and Flow Charting	Microsoft Visio	2003
Email	Microsoft Outlook	2003
Project Management	Microsoft Project	2003
Database	Microsoft Access	2003
Desktop Publishing	Microsoft Publisher	2003
Anti-Virus	Symantec Antivirus	10.x
Screen Capture	Gadwin Print Screen	4.0.1

NOTE: Administrative security will not be given to the project machines if they are owned by the Commonwealth of Massachusetts. All machines must be kept compliant with ITD policies for security, Virus scanning, and other policies.

2.2 Servers

Purpose	Standard	Version
Operating System(s)	Windows Server	
	RedHat Linux	
Web Server(s)	Internet Information Server (IIS)	
	Apache Web Server	
Database Server(s)	Microsoft SQL Server	

EOHHS CTO Organization	Version: 1.0
Secretariat Application Reference Manual	Date: February 02, 2011

	Oracle	
Application Server(s)	Internet Information Server (IIS)	
	Oracle WebLogic	
	JBoss	

2.3 Telephony

Purpose	Standard	Version
Telephone	Cisco IP Phone	7900 Series
Call Processing System	Cisco Unified Communications Manager	6
Voice Mail Management System	Cisco Unity	5
Mobile Access	Cisco Mobility	
Rich Media conferencing	Cisco Meeting Place	5

2.4 Assistive Technology

Purpose	Standard	
Low Vision	Zoom Text	
	Phone + Tenacity AccessAPhone	
Blind	JAWS	
	Phone + Tenacity AccessAPhone	
Hard of Hearing	Handset Amplification	
	Phone + Inline Amplifier	
Deaf	Unity TTY Tools	
	Phone + FX Port + Unity TTY Tools	
Physical Impairment	Dragon Naturally Speaking	
	Phone + IP Communicator + Tenacity AccessAPhone	

2.5 Security

Security standards adopted by ITD must be followed. More information is available at:

<http://www.mass.gov/?pageID=afsubtopic&L=5&L0=Home&L1=Research+%26+Technology&L2=IT+Policies%2c+Standards+%26+Guidance&L3=Enterprise+Policies+%26+Standards&L4=Security+Policies+%26+Standards&sid=Eoaf>

2.6 Information Technology Acquisition Policy

Information technology acquisition policy adopted by ITD must be followed. More information is available at:

http://www.mass.gov/?pageID=afterminal&L=4&L0=Home&L1=Research+%26+Technology&L2=IT+Policies%2c+Standards+%26+Guidance&L3=Enterprise+Policies+%26+Standards&sid=Eoaf&b=terminalcontent&f=itd_polic

EOHHS CTO Organization	Version: 1.0
Secretariat Application Reference Manual	Date: February 02, 2011

[ies_standards_it_acquisitionpolicy_2dot0&csid=Eoaf](#)

2.7 Desktop Power Management Policy

Desktop power management policy adopted by ITD must be followed. More information is available at:

http://www.mass.gov/?pageID=afterterminal&L=4&L0=Home&L1=Research+%26+Technology&L2=IT+Policies%2c+Standards+%26+Guidance&L3=Enterprise+Policies+%26+Standards&sid=Eoaf&b=terminalcontent&f=itd_policies_standards_power_mgmt_standards1dot0&csid=Eoaf

2.8 Open Standards Policy

Open Standards Policy adopted by ITD must be followed. More information is available at:

http://www.mass.gov/?pageID=afterterminal&L=4&L0=Home&L1=Research+%26+Technology&L2=IT+Policies%2c+Standards+%26+Guidance&L3=Enterprise+Policies+%26+Standards&sid=Eoaf&b=terminalcontent&f=itd_policies_standards_open_standards_policy&csid=Eoaf

EOHHS CTO Organization	Version: 1.0
Secretariat Application Reference Manual	Date: February 02, 2011

3. Website Information Architecture Services

3.1 Access Channels

3.1.1 Portal

The secretariat web site is integrated into the mass.gov portal. It is mandatory that the agency specific websites are integrated into the same portal. This ensures secure, reliable websites and provides a single face for all government agencies. All the sites must have the same look and feel as the mass.gov portal.

3.1.2 Virtual Gateway

All web applications must be deployed in and accessed via the Virtual Gateway. This will allow various applications to be accessed uniformly via the URL
<https://service.hhs.state.ma.us/<application name>>.

3.1.3 XML Gateway

All business web services provided by the agency must be accessed using the XML gateway.

3.1.4 IVR

The guidelines are being evaluated and will be published in future.

3.1.5 Mobile Devices

Support for mobile devices in their native mode is not available. However mobile device that have browser capabilities can be used to access the applications.

The guidelines are being evaluated and will be published in future.

3.2 Web Site Policies & Guidance

Policies and best practices for creating web sites, adopted by ITD, must be followed. More information is available at:

Website Policies: www.mass.gov/itd/webpolicies

Website Guidance: www.mass.gov/itd/webguidance

3.3 Accessibility

Accessibility standards for developing web content and web applications, adopted by ITD, must be followed. More information is available at:

Accessibility Standards: www.mass.gov/itd/accessibilitystandards

EOHHS CTO Organization	Version: 1.0
Secretariat Application Reference Manual	Date: February 02, 2011

4. Application Services

4.1 Project Management

4.1.1 Methodology

The development methodology adopted by the PMO office must be followed. This methodology provides guidelines for application development and is based on Rational Unified Process. More information is available at:

Project Methodology and Life Cycle Guide: <https://massforge.state.ma.us/sf/go/doc35367?nav=1>

4.1.2 Collaboration

MassForge is EOHHS' enterprise web-based collaborative project management system, empowering team collaboration and centralization of project activities and artifacts. It provides versioning of the documents and controlled access.

More information is available at: <https://massforge.state.ma.us/sf/sfmain/do/home>

4.2 Enterprise Services

This section lists out the services that are provided at the secretariat level. These address the common requirements for various applications. These services should be leveraged by the agency applications.

4.2.1 Access and Identity Management Services (AIMS)

Access and Identity Management Services provide shared security services for applications deployed in virtual gateway and services deployed in the integration gateway. These services include authentication, authorization and single sign-on. They ensure that user access is consistent with policy and regulatory requirements. It manages the identity of the end user, roles and policies that grant access to resources.

All applications hosted in virtual gateway and services hosted in integration gateway must be integrated with AIMS.

More information is available at: <https://massforge.state.ma.us/sf/go/page1405>

4.2.2 Enterprise Document Management (EDM)

EDM is the shared document scanning and back-end repository system for retrieving/storing unstructured content (images, mixed media, emails, etc). EDM provides the following services for use by agencies:

1. Document Scanning Solution: Digitizes the documents and stores them in the repository.
2. EDM Web Service: Allows programmatic access to the repository.
3. Collaborative Management: Workflow solutions for managing content.
4. Content Archival: Provides retention and disposition capabilities.

EOHHS CTO Organization	Version: 1.0
Secretariat Application Reference Manual	Date: February 02, 2011

All agencies with applications dealing with unstructured file content must use EDM services.

More information is available at <https://massforge.state.ma.us/sf/go/page1406>

4.2.3 XML Gateway

XML Gateway provides a centralized management for virtualized business services. It provides security, policy evaluation, transformation and routing to business services. It utilizes hardware based features for improving performance. All agencies must use XML Gateway to provide access to business services.

More information is available at <https://massforge.state.ma.us/sf/go/page1416>

4.2.4 File Transfer Services

The FTS provides HHS business services the ability to exchange files among each other and /or with external business partners. It also allows end users to share files with each other. The facilities can be utilized by integrating the web service calls into the application or by using the web based user interface. It supports both pull and push models for exchanging files between business services. A user/application can "publish" a file to one or more "destinations". Each file within the FTS Service is tagged with an "expiration date"; once a file expires it will no longer be available to the FTS Users. All agencies that need to share files with multiple destinations and in need for a visual screens to manage files must use FTS.

NOTE: This might be replaced by Interchange

More information is available at <https://massforge.state.ma.us/sf/go/page1412>

4.2.5 Enterprise Service Bus

ESB provides the business integration software infrastructure that enables communication between applications and services - i.e., applications integration and/or custom adapters for inter-agency messaging/data flow solutions. The ESB Core provides Asynchronous or Synchronous messaging for applications and services. The ESB communicates across multiple integration end points such as databases, files, queues, web services or FTP. ESB "Adapters" can "pull" data from sources and "push" data to the next end point. The ESB validates, transforms, routes, and delivers the data/messages.

More information is available at <https://massforge.state.ma.us/sf/go/page1408>

4.2.6 Virus Scan Service

Virus Scan service provides a mechanism to scan a file for viruses. All applications that support file uploads must use this service to scan for viruses before the application can process the file.

More information is available at

EOHHS CTO Organization	Version: 1.0
Secretariat Application Reference Manual	Date: February 02, 2011

https://massforge.state.ma.us/sf/docman/do/listDocuments/projects.exa/docman.root.type.1_wsd1.virus_scan_ws

4.2.7 HHS Reporting Service

Cognos based HHS Reporting service provides a centralized mechanism for users to find reports, schedule execution of report and provide notifications. It supports pre-formatted and ad-hoc reports based on relational or OLAP data sources. All agencies that need adhoc, slice and dice reports must base their solution on these centralized services.

4.2.8 HHS Interchange

Interchange will provide agencies the capabilities of managed file transfer in a secure environment for transferring sensitive data files securely between them.. The capabilities are available for person to person, person to system and system to person and system to system transfers. All agencies must use interchange to transfer data files.

More information is available at https://massforge.state.ma.us/sf/projects/hhs_interchange

4.2.9 Log Services

Log services offer a centralized management of log files. The log files from different deployments are moved to a central location on a schedule. The applications support teams can access these logs without the support from production support teams.

For details on the Virtual Gateway log solution, please review this link:
<https://massforge.state.ma.us/sf/go/doc22738?nav=1>

4.2.10 Application Monitoring

Application monitoring services are available as a shared service. The monitoring capabilities are supported by the product Wily Introscope. Application teams that need to identify performance issues can leverage these services to monitor their applications. There are also Commonwealth-wide monitoring services. This section should appropriately identify when to use Wily and when to use the enterprise services.

4.2.11 Address Verification

The Commonwealth already has an application that performs address verification as part of the NewMMIS project. The address verification application is built on top of the US Postal Service application which is updated monthly. The application exposes its services via a Web Service API. Any applications that need to do address verification should consult with the Enterprise Team for more information about how they can leverage the existing solution.

4.2.12 Geographical Information Services (GIS)

Additionally, the Commonwealth has a solution for determining the geographical coordinates of an address. The coordinates can they be used to find other locations in the general vicinity. In the NewMMIS project, providers "near" the member can be found as an example. Please consult the Enterprise Team for more information.

EOHHS CTO Organization	Version: 1.0
Secretariat Application Reference Manual	Date: February 02, 2011

4.2.13 Quality Assurance Services

Quality Assurance team offers quality assessment services. Performance testing, load testing, capacity planning, accessibility testing, usability testing and vulnerability testing services are offered. Application teams should leverage these services before an application is deployed in production.

4.2.14 Release Management

Release management team offers services to maintain the application in the 5 different environments used in HHS. These services range from build, deploy, assist in trouble shooting of application issues.

4.2.15 Backup and Recovery

Backup and recovery services are offered by ITD. Multiple service levels and configuration options are offered that can be tailored for the specific business needs. More information available at:

http://www.mass.gov/?pageID=afterterminal&L=5&L0=Home&L1=Research+%26+Technology&L2=Information+Technology+Services+%26+Support&L3=The+Information+Technology+Division's+Services+Catalog&L4=Network+%26+Data+Services&sid=Eoaf&b=terminalcontent&f=itd_services_services_catalog_nad_backup_recovery&csid=Eoaf

4.2.16 HL7 Gateway Service

HL7 gateway service offers a centralized gateway to exchange information between external and internal systems using HL7 format. It offers centralized management of security and data policies, translation into canonical format, rule based routing to internal systems.

More information is available at: <https://massforge.state.ma.us/sf/projects/hl7gateway>

4.2.17 Sybase Translator Service

The Healthcare Transaction Services (HTS) Web Service is a web service that will enable external interface partners to perform healthcare related EDI operations in synchronous or asynchronous mode. More information is available at: <https://massforge.state.ma.us/sf/go/doc4879?nav=1>

4.2.18 WTX Service

WTX service offers a centralized translation service for different data formats. Applications that need to transform incoming data for processing can leverage this service.

More information available at: <http://massforge.state.ma.us/sf/go/doc18481>

4.3 Architecture

Due to the nature of the business HHS is involved in, as well as the organizationally and functionally distributed business environment, Service Oriented Architecture is the standard approach. It allows seamless integration of the multitude of legacy applications in HHS and a consistent way to meet the quality of service requirements. It provides building blocks for combining simple services into fully featured, value-added services and achieves integration and rapid development of new business offerings. The location-independent and functionally-decoupled attributes of a service-oriented architecture will allow flexibility in how applications are deployed.

EOHHS CTO Organization	Version: 1.0
Secretariat Application Reference Manual	Date: February 02, 2011

All applications must follow the set of architectural principles defined below.

4.3.1 Architectural Principles

Think business process/service, not system/application

Orient service development around business processes so that it is possible to execute these processes from the business models directly through a process engine and also measure, monitor and administer processes dynamically

Adopt model-driven architecture

Insulate application architecture from the underlying technology platform by moving in the direction of a model-driven architecture.

Define domain frameworks

Adopt domain frameworks so that new features can be added by using a few new components with minimal changes to existing code

Use hybrid approach for legacy application management

Adopt a hybrid approach for legacy application management involving both componentization and encapsulation of existing functionality, and integration of conformant systems and services.

Design for service quality

Use workload models based on various factors to scale systems in order to meet present and anticipated Quality of Service (or Service Level) requirements.

Build for collaboration

Adopt technologies to facilitate inter-enterprise collaboration and process automation.

Adopt open and standards-based design

Adopt process and service standards to harmonize processes of the business domain and streamline interactions

Treat data as a shared asset

Adopt technologies that allow data to be shared consistently across the enterprise, with relevant users having access to the data necessary to perform their duties

Secure information

Implement security measures to protect information at the network, container, and service (code) levels in order to provide for the Confidentiality, Integrity, and Availability of the data.

4.3.2 Quality of Service

The above principles guide the technical architecture of information systems and applications that ultimately help realize the business drivers. Hence the architectural principle should satisfy one or more Quality of Service parameters (QoS) that help realize the strategic priorities and business drivers. The following QoS factors have been identified based on industry research and the architectural principles covered in the previous section:

EOHHS CTO Organization	Version: 1.0
Secretariat Application Reference Manual	Date: February 02, 2011

- Performance – ability to deliver results (throughput or bandwidth) within the least response time (latency)
- Scalability – ability to cater to greater demands imposed upon the system (e.g.: support increased number of users, products) without affecting any of the other QoS parameters
- Reliability – ability to function with the least occurrence of failure
- Availability – ability to maximize the time when the system is available for use
- Security – ability to authenticate and authorize users to provide secure access to the system in a traceable (auditable) manner
- Manageability – ability to monitor and configure systems easily and detect operational characteristics related to performance and failures (remotely).
- Maintainability – ability to modify the system easily, with the minimum amount of work or rework over the life cycle of the application
- Extensibility – ability to make significant enhancements or changes easily.
- Usability – ability to allow users to use and navigate the system easily.
- Serviceability – ability to be repaired or updated easily and rapidly without affecting reliability or availability of the system.
- Reusability – ability to use individual components or services in the building of unrelated modules or services
- Interoperability - ability of components to work with each other regardless of their underlying platform.

The architecture is described through a set of views, each from the perspective of a different stakeholder. An individual view captures items meaningful to the stakeholders as elements and their interrelationships expressed in a standard form, the structure of the view, and the view's correlation with other views.

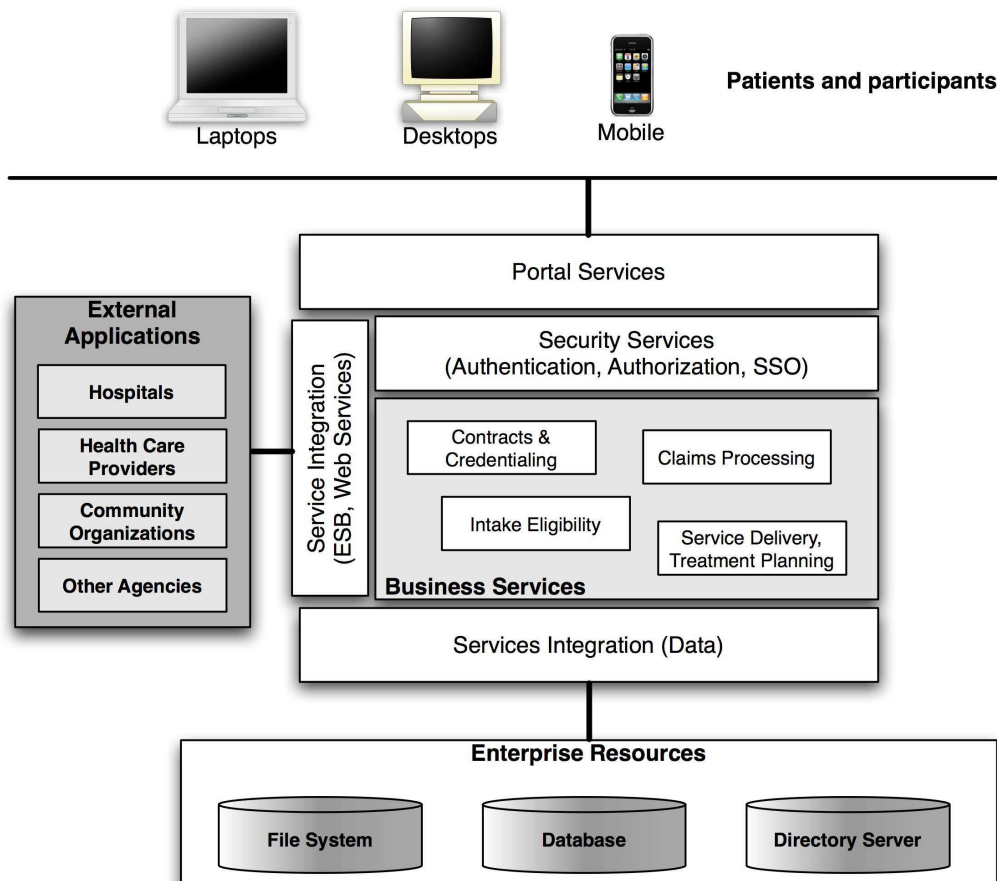
4.3.3 Conceptual View

The Conceptual View serves to highlight the distinguishing concepts of the architecture. As illustrated in the Conceptual View figure, the architecture is composed of a collection of functionally decoupled business services residing within a unified architecture. They encapsulate disparate business functionalities yet share a common substrate and semantics. The business services may retrieve and integrate information residing in backend enterprise resources such as databases and legacy applications through a common Service Integration layer. The Service Integration layer standardizes the method and the semantics of accessing the enterprise resources. Once the business service has retrieved the information from the enterprise resources, they can aggregate this information and expose the results as a network accessible service to the users. These services are then delivered to the users through a common delivery channel regardless of the type of client device. This common Service Delivery channel provides a single point of access to the business services and ensures that centralized security and context policies are applied to each user.

EOHHS CTO Organization	Version: 1.0
Secretariat Application Reference Manual	Date: February 02, 2011

Since they are decoupled from one another, services can be deployed in whatever configurations necessary to meet business needs. Further, as business requirements evolve, so can the business services, as new services can be rapidly assembled instead of being developed or purchased.

A unified, service oriented architecture with a centralized delivery channel and a common integration layer enables the realization of consistent quality of service across the enterprise. Aspects such as scalability, security, availability, reliability and maintainability can be readily predicted, easily controlled and managed.



Conceptual View of the architecture

4.3.4 Logical View

In the Logical View, the architecture is decomposed into parts that address the different functionality areas. It also describes the relationship among these parts and how they interact with one another.

4.3.4.1 Overview - Taxonomy of IT Services

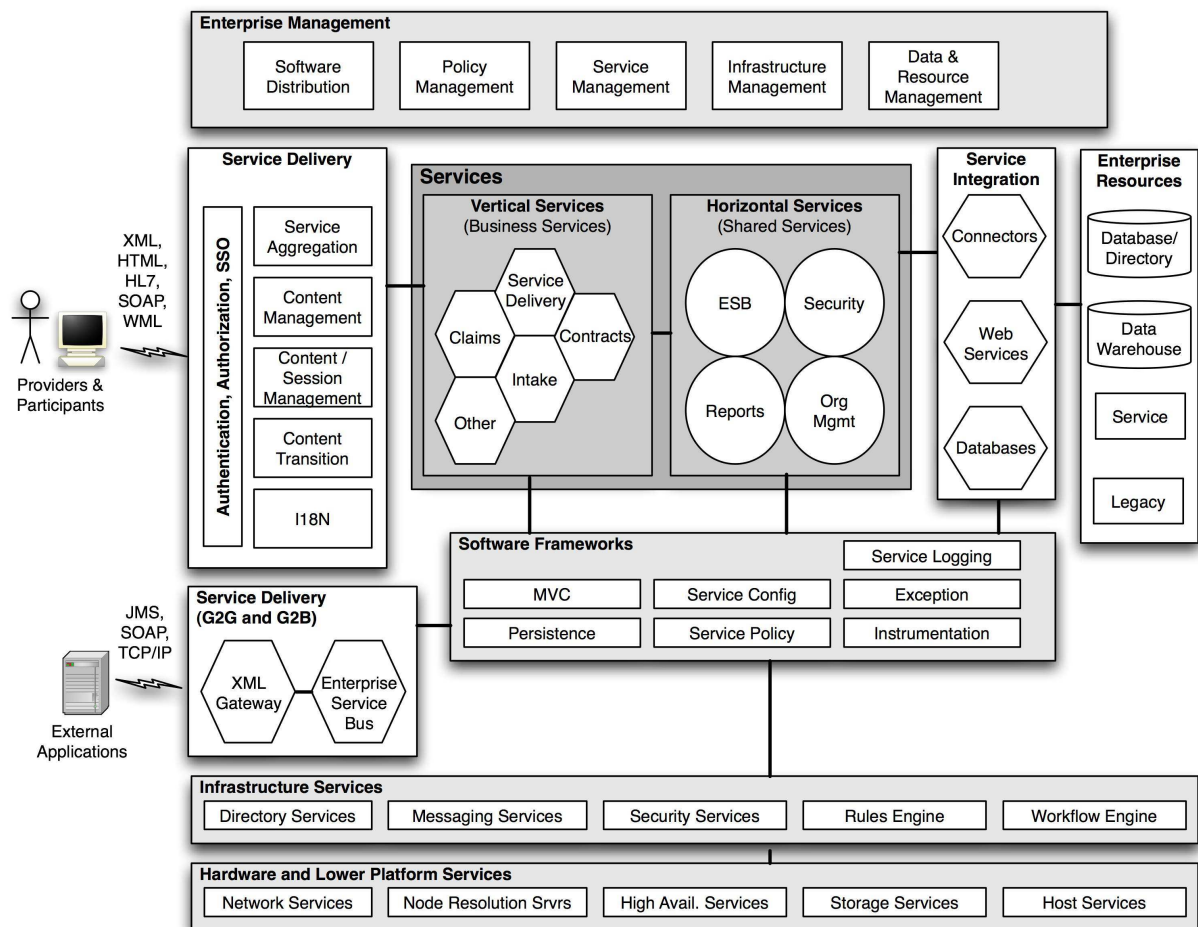
The organization and categorization of a set of unrelated objects is known as taxonomy. It is convenient to group services by their functional area, and at the highest level the initial set of

EOHHS CTO Organization	Version: 1.0
Secretariat Application Reference Manual	Date: February 02, 2011

groups (services taxonomy) is as follows:

- Frameworks
- Service Delivery
- Business Services
- Infrastructure Services
- Service Integration
- Hardware and Lower Platform
- Enterprise Management

The figure below shows how these service groups are related and highlights the significant or representative services within each group.



EOHHS CTO Organization	Version: 1.0
Secretariat Application Reference Manual	Date: February 02, 2011

4.3.4.2 Frameworks

Frameworks are generic, typically business-agnostic reusable components and libraries. The core function of the business services is to deliver some business functionality. However, these business services must also perform some common, non-business-specific tasks, for example to provide case management functions, manage the Software Development Life Cycle, and initialize a database connection or to perform monitoring and management functions. The extraction of these common functionalities from the services and utilizing them as a shared framework across the services increases architect, designer, and programmer efficiency. A shared framework also offers consistent and predictable quality of service throughout the system. Widespread use of common software frameworks imposes a common programming style across applications, enabling greater programmer mobility within the enterprise. Frameworks can also improve portability and protect the application components from instability in the upper platform layer (i.e., vendor specific Web servers and application servers). When the frameworks are developed based on the capabilities of standards-compliant virtual platforms (e.g. Servlets, JSP, and EJB), changing from one vendor to another will not have dramatic impacts, particularly if the frameworks are designed to shield vendor specific implementation details.

This set of frameworks address basic functionality that will be needed by all initiatives and are discussed briefly in subsections below. The frameworks are categorized into Application and Infrastructure frameworks.

4.3.4.2.1 Application Frameworks

Application frameworks are independent of the infrastructure platforms while providing non-business specific functionality.

Service Logging

Achieving manageability goals requires producing an accurate trace of application activity in the form of a streaming log. These logs are used for operations management, auditing, and troubleshooting. They typically capture information such as security failures, configuration errors, performance information, and bugs encountered in the application or platform. Well-designed and consistent logging facilitates software servicing and maintenance at customer sites by producing log reports suitable for analysis by end users, system administrators, field service engineers, and software development teams.

Service Configuration

Every application requires customization at some level to indicate desired settings for variant facets. Examples include directory locations, switch settings, level selections, and component lists. A common declarative approach to storing and accessing these variant facets based on XML configuration files will ensure a level of consistency for application configuration and deployment. Further, the use of XML enables the use of industry-standard tools for transformation and manipulation of application configuration files.

Service Exceptions

EOHHS CTO Organization	Version: 1.0
Secretariat Application Reference Manual	Date: February 02, 2011

One important issue that must be addressed by all Java programmers is exception handling. The ITA must establish an enterprise-wide approach to exception representation and processing to avoid duplication of effort between application groups, and also to ensure a consistent level of response to operational problems. That is, exception handling is typically given short shrift by designers and programmers because it is time consuming and difficult to handle well. Providing exception handling in the form of an enterprise-wide framework will mitigate the risk that application groups will leave it to the end of the design and development cycles, avoiding the concomitant danger that exceptions are not handled at all. Furthermore, exception handling will be integrated with the logging and event frameworks for recording and notification.

Events and Messaging

A common requirement for applications is enterprise-level messaging. When something happens in one area, other areas may need to be informed of it, sometimes across application boundaries. The notification can be propagated synchronously or asynchronously depending on the circumstances. Further, the originator of the event may or may not need to know the receiving parties.

Service Registration and Discovery

Services must expose themselves to the world in some way. The standard way of doing it is to publish their availability in a directory server, analogous to yellow pages. Obviously, for ease of management and development, the fewer directory servers there are and the more transparent they are, the easier it is for system administrators and developers. Service registration can be accomplished through established standards such as Lightweight Directory Access Protocol (LDAP) via JNDI. In the future this may evolve to include emerging standards such as Universal Description, Discovery, and Integration (UDDI). In both cases, Java APIs exist for accessing the underlying mechanisms (JNDI for LDAP and JAXR for UDDI).

Service Instrumentation

A number of solutions for systems management are available, based on standards such as Simple Network Management Protocol (SNMP), Desktop Management Interface (DMI), and Common Management Information Protocol (CMIP), as well as other more proprietary protocols. In 1996, the Internet Distributed Management Task Force initiated the Web-Based Enterprise Management (WBEM) effort, envisioned as a single web-based standard that integrates the other standards. WBEM and closely related Common Information Model (CIM) are now well established as DMTF standards, and most major vendors are adapting their solutions for compatibility.

In the Java community, Java Management Extensions (JMX) is being developed as a common API for accessing management information spanning from the lowest level hardware component to the proprietary application components. Defined on top of standards such as WBEM/CIM, JMX provides the tools for building distributed Web-based, modular and dynamic solutions for managing devices, applications and service-driven networks.

Model-View-Controller

The idea behind Model-View-Controller has its roots in the Smalltalk language environment. The goal of this framework is to achieve decoupling among the software components that are responsible for encapsulating business functions, rendering the content and controlling the navigation or flow. This fundamental concept can be applied to a collection of Servlets, JSPs, JavaBeans and EJBs to achieve

EOHHS CTO Organization	Version: 1.0
Secretariat Application Reference Manual	Date: February 02, 2011

architecturally decoupled J2EE systems. JSPs are used to represents Views, Servlets as Controllers and JavaBeans and EJB's as the Models.

4.3.4.2.2 Infrastructure Frameworks

System frameworks perform low-level functionality on behalf of the caller and interact with the infrastructure services to realize the request. They serve as bridge between the business service and the infrastructure service.

Service Policy

At least two aspects of security are common across all applications. As users access the services, they must be accurately identified (authentication) and their permitted access levels must be effectively managed (authorization). Using a common framework for these two security activities will reduce the burden on application development teams.

Session Management

Session management manages the state of an application as a server-side object. Session management deals with clients and the conversational state that is built up in the course of a conversation. It must be kept in an appropriate location, based on the requirements of the application in terms of whether a user can transparently re-establish their context with the system. Thus, there are a variety of options. A framework should handle each option that needs to be supported, so that it is done once and only once. Persistence deals with storing state with guarantees of integrity, consistency, and durability.

Transaction Management

A service driven architecture must address the familiar information technology problem of transaction management to ensure the atomicity, consistency, isolation, and durability (ACID) requirements of applications. The Java Transaction API (JTA) of J2EE provides a solid basis for a common transaction management framework. J2EE also includes the feature to delegate the responsibility of transaction management to the underlying J2EE platform, which allows only the business logic to be included in the application components.

Persistence

Business objects manipulated by services must be procured and stored in permanent data storage, usually realized as relational databases. A common approach to object persistence using either EJB Container Managed Persistence (CMP) or a third-party persistence mechanism must be selected. If persistence is required in the Presentation Tier for sites that rely only on Servlet and JSP, a common data source location and resource-pooling framework is required.

Directly related to Object Persistence, caching of frequently accessed, read-only persistent objects (e.g. code tables, option descriptions) at the application server level can greatly enhance application performance. A common framework for loading and refreshing these reference caches will ensure that application teams can utilize this performance enhancement

Service Context

Service Context is adequately detecting and managing the client device profile to enable proper presentation of service results. Therefore, whether a consumer is accessing a Web service from a

EOHHS CTO Organization	Version: 1.0
Secretariat Application Reference Manual	Date: February 02, 2011

WAP-enabled phone, MIDP hand-held device, Internet browser, or rich client should not affect any service processing other than the presentation layer.

4.3.4.3 Service Delivery

The information the Business Services provides to its users must be translated, formatted and delivered based on the capabilities of the device the user utilizes to access the system. Service delivery addresses the way in which services are presented to users. Services can be tailored to a specific user, based on a number of factors, such as the device they are using for access, their location, their indicated preferences, or the type of user they are. All of this is done in conformance with the designated policies. Further, services from the enterprise may need to be aggregated into larger units, presenting the user with a diversity of services on demand.

4.3.4.4 Business Services

Services encapsulate the high-level business functionality and offer these capabilities to the users and other services. In the HHS architecture, services are categorized as Vertical (or Business) Services and Horizontal (or Shared) Services. The distinction is made depending on the scope of the service. Horizontal service is a micro-service that performs a specific functionality related to infrastructure. Vertical service is a business component that represents a “macro-service” such that it provides a complete set of functionality related to business processing (E.g. Intake Eligibility and enrollment). Vertical service could be a coarse-grained component, which talks to other business components to form a functional public service.

Horizontal Business Services

Horizontal Business Services do not directly offer business functionality to end-users. Rather, they are meant to encapsulate common business functionality that may be present across different vertical business services. Hence, horizontal business services are not “owned” by a specific business unit but instead are shared by many organizations.

Vertical Business Services

Vertical business services encapsulate core, high-level business functionality such as Contracts and Credentialing, Claims Processing, Eligibility, Enrollment and Service Delivery that are offered by the various organizational units within HHS.

4.3.4.5 Infrastructure Service

Infrastructure services provide system level support services for the business services. They are the foundational mechanisms of technology that allow other services to be built. Typically, they address areas such as directory interaction, security, messaging and state management.

Directory/Meta Directory Service

The Directory/Meta directory service stores enterprise data in a hierarchical relationship, typically to support read only operations such as for authenticating and/or authorizing against stored user credentials.

Messaging Service

The Messaging Service provides the core queue infrastructure and programming extensions (JMS API support) to be able to provide asynchronous capabilities in the application architecture.

EOHHS CTO Organization	Version: 1.0
Secretariat Application Reference Manual	Date: February 02, 2011

Security Service

The Security service uses the directory/meta directory and policy services to authenticate and/or authorize users. It also collects the entitlements based on the credentials stored in the policy server to provide fine-grained security features in the application architecture.

Email Service

The Email service provides a centralized mechanism for exchange of information through the sending and receiving of mail messages. Some of the collaboration services like scheduling meeting, appointments and calendar services can also be accomplished using email service.

Transaction Management Service

The Transaction management service handles transaction life cycles to a particular resource like database or an ERP system and across multiple resources, which might involve complex scenarios like two phase commits. The J2EE application server provides this as an out of the box functionality.

Resource Management

Resource management service includes pooling a set of costly resources like Database connections, instances of business service component, threads and allocating them based on request/demand. It is a very critical part of any application architecture as the management of these resources has a very large impact on the Quality of Service (QoS) goals of the system. J2EE compliant application servers are mandated to provide this functionality out of the box.

Application Clustering Service

Application clustering service involves distributing the application instances on multiple hardware platforms having the same infrastructure set up. The Clustering service aims at balancing the load on the system by distributing it horizontally across instances that are tied to together as a cluster so that they share their state and session information. J2EE application servers support and allow clustered mode of deployment and there are standard best practices that have been developed to be followed for components that are deployed in clustered mode.

Persistence Service

Persistence service provides the access layer over the data resources like the database products for the business service components that are deployed in the J2EE application server. Persistence service can be custom developed or realized using COTS products.

Rules Engine

Rules engine extracts the business specific information and stores it in a configuration that is referenced by the business services. They move the logic driven rules to configuration from code so that when business drivers change it must be a matter of altering the rules through the rules engine and not the code.

Workflow Engine

Workflow Engine is the Process manager component that handles the decision flow and process flow mechanism between services. Most common application server vendors provide a process management

EOHHS CTO Organization	Version: 1.0
Secretariat Application Reference Manual	Date: February 02, 2011

module that lets you design the decision logic that flows through the business service components deployed in the application server.

4.3.4.6 Service Integration

Business services access enterprise data, interact with COTS packages and invoke legacy applications in order to fulfill the user's request. Service integration provides the business services with the capability to interface with these Enterprise Resources using mechanisms such as a message bus, custom and standard compliant connectors.

Custom Connectors

Custom connectors are components or a collection of components that provides a set of proprietary or custom developed APIs for accessing Enterprise Resources such as ERP, CRM or legacy systems.

Messaging

Messaging in the context of service integration refers to a collection of components that provide a set of APIs for the synchronous and asynchronous transfer of data to external business partners or to Enterprise Resource such as legacy systems.

Standard Connectors

Standard connectors refer to components or a collection of components that provide a set of APIs to communicate with resources that expose standards compliant interfaces such as Java Database Connectivity (JDBC) and J2EE Connector Architecture (JCA).

Hardware and Lower Platform

Underlying all of the other service taxonomies, Hardware and Lower Platform provides the processing, storage, and network hardware, as well as the operating system and networking protocols.

Networking Service

Network services provide the capability to connect, transfer, transform, and route hosts or clients on LANs/WANs in order to support information exchange. Transfer data types can include Wireless, Voice, and Video.

Name Resolution Service

Name resolution services provide host and data name resolution for network resources. They include services such as DNS, WINS, and TNS Names Server.

High Availability/Performance Service

This category of services provides high availability and performance improvements by distributing access and load of hosts and/or clients. Includes services such as content switches, content engines, hardware based SSL accelerators, and network load balancers.

Storage Service

Storage services provide random and/or sequential data access. They include services such as SAN, NAS, disk subsystems, tape subsystems, optical disk subsystems, and flash memory devices.

EOHHS CTO Organization	Version: 1.0
Secretariat Application Reference Manual	Date: February 02, 2011

Host Services

Host services include the hardware platforms on which all of the IT services and Enterprise Resources reside.

4.3.4.7 Enterprise Management

All aspects of an enterprise must be managed to ensure its smooth operation. Enterprise management includes all of the activities related to the monitoring and management of enterprise assets including the automated, network enabled distribution of applications and components. Policies for access, authentication, logging and other areas must also be managed. Business Services and infrastructure mechanisms must also be monitored for availability and performance. The same is also true for Enterprise Resources.

Release Management

Software distribution service provides the capability to manage the dissemination of executable code.

Security Policy Management

Security policy management provides the capability to monitor and enforce the adherence of authentication, authorization, encryption and auditing guidelines and best practices across the enterprise.

Service Management

Service management provides the capability to monitor and administer application level business services.

Infrastructure Management

Infrastructure management provides the capability to monitor and administer hardware platforms as well as web and application servers, messaging servers and other virtual platforms.

Data Management

Data management provides the capability to monitor and administer databases and directory servers.

Network Management

Network management provides the capability to monitor and administer networking hardware such as routers, bridges and switches as well as storage devices.

Backup/Recovery

This category of service provides the capability to perform redundancy operations and restoration of data.

Configuration Management

Configuration management provides the capability to perform version control and the tracking of changes to resources such as source code and project documents.

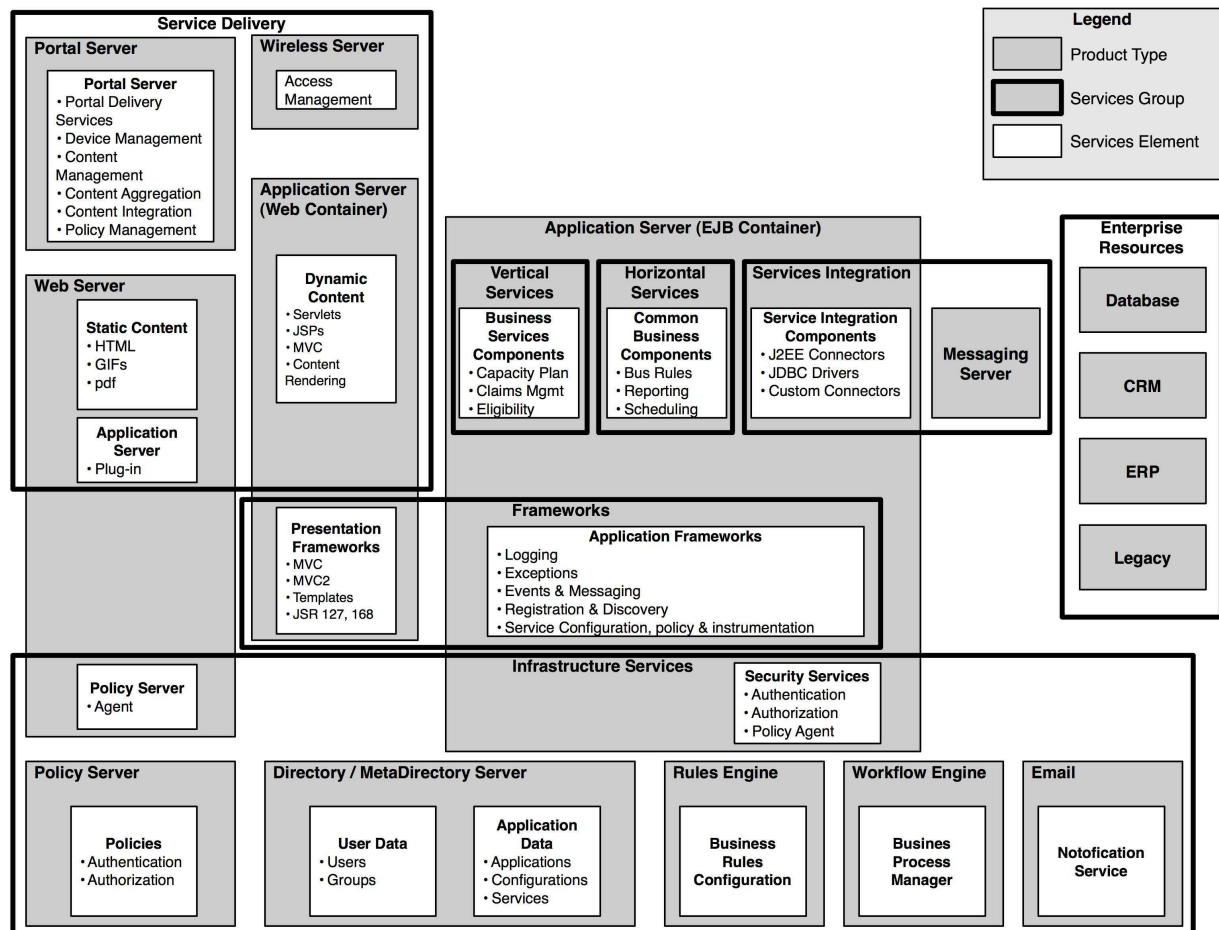
EOHHS CTO Organization	Version: 1.0
Secretariat Application Reference Manual	Date: February 02, 2011

4.3.5 Process View

This section elaborates on each of the elements of the ITA and maps them to the process, or execution environment, that they will run in. As indicated, each of the Taxonomy groupings consists of several elements that may span multiple processes.

As shown, the main processes will be:

- Web Server
- Application Server
- Directory Server
- Policy Server
- Messaging Server
- Database



EOHHS CTO Organization	Version: 1.0
Secretariat Application Reference Manual	Date: February 02, 2011

4.3.5.1 Web Server

The Web Server is the infrastructure that handles the following

- Serving of static HTML pages
- Serving image files (JPG/GIF)
- Serving document files (PDF)

The Web Server infrastructure also comes in with a plug-in to invoke the Servlet/JSP container that resides in the infrastructure of the J2EE Application server or the Web server to handle presentation tier functionalities.

4.3.5.2 J2EE Application Server

The J2EE Application Server is the platform designed for enterprise customers and independent software developers who require high quality of service and performance for mission critical e-commerce applications. It is used to build and deliver application services to a broad range of servers, clients and devices and to integrate with business-to-business applications.

The J2EE Application Server offers a host of value added features that extend the base capabilities and services offered by supporting the standard APIs without compromising the basic standards compliance customers rely on. They provide support for Enterprise Java Beans, Servlets, Java Server Pages, Java Mail, Java Activation Framework, Java Transaction API, Java Transaction Service, Java Messaging Service, Java Naming and Directory Service, JDBC, RMI/IIOP, Corba, Java API for XML Processing, and Web Services. The two major components of a J2EE application server are the EJB container and Web Container.

Web Container

The Web Container provides support for presentation components like Java Servlets and JSP. Servlets act as the Controller that manages the incoming HTTP requests and JSPs are used for the content aggregation and presentation. The Web container also hosts open standards based frameworks like Apache Struts and MVC components.

EJB Container

The EJB Container provides the basic infrastructure and support for hosting the Enterprise Java Beans (EJB) that perform most of the business functionality in J2EE environment. They also provide lifecycle management of these components and manage transactions, security, and persistence. EJB containers also provide the environment for integration with external legacy applications like the ERP, CRM and Mainframes by supporting the J2EE Connector Architecture (JCA).

4.3.5.3 Policy Server

The Policy server infrastructure provides storage capability for user/roles information and the eligible entitlements for the respective roles. This information will be used to provide “fine-grained” access control to the various business and application functionalities at a component level in the J2EE platform. The role information stored in this infrastructure can be propagated across tiers and hence a consistent implementation of security could be achieved.

4.3.5.4 Directory/Meta Directory Server

The Directory Server is the infrastructure that provides a central repository for storing and managing identity profiles, access privileges and application and network resource information. Information stored in the Directory Server can be used for the authentication and authorization of users to enable secure

EOHHS CTO Organization	Version: 1.0
Secretariat Application Reference Manual	Date: February 02, 2011

access to enterprise and Internet services and applications. It helps improve security and protection of key corporate information assets by ensuring appropriate access control policies are enforced across all communities, applications, and services on a global basis.

4.3.5.5 Messaging Server

The Messaging Servers provide the core infrastructure for managing messages and the organization of message destinations like Queues/Topics and its associated messages. They also provide access APIs for interacting with core messaging infrastructure. In the J2EE platform, Java Message Service (JMS) is the standard API for the Java components to interact with the messaging server. Most popular messaging servers like SonicMQ, Sun ONE MQ, and IBM MQ Series provide support for JMS to be able to send and receive messages.

4.3.5.6 Database

Database is the infrastructure that enables storage of application and business information. It resides in the resources tier in the J2EE platform as a repository for storing all business related data. Databases provide drivers which act as the mechanism to access them externally. In the J2EE platform the JDBC API defines the access mechanism for the databases. Most of the commercial vendors such as Oracle, Sybase and IBM DB2 provide JDBC support.

4.3.5.7 Workflow Engine

Workflow Engine is the infrastructure that provides process automation capabilities in the J2EE platform. The workflow engine helps in designing the process flow between the business components using standard GUI. In the J2EE realm, most of the business components are modeled as services and the workflow engine manages the informational and logical flow between these services to accomplish a complete business process. Some widely used workflow engines include Weblogic process manager and Versata.

4.3.5.8 Rules Engine

Rules Engine is the infrastructure in the J2EE platform that enables the components to be configured according to the business needs. Enterprise Java Bean components that perform most of the business functionality can be configured to base business operations according to rules specified through a rules engine. Most of the common rules engines have rules editor to input rules and XML based rules configuration and access APIs to programmatically read the rules. Some of the most popular rules engines are ILOG JRules and Advisor Blaze rules engine.

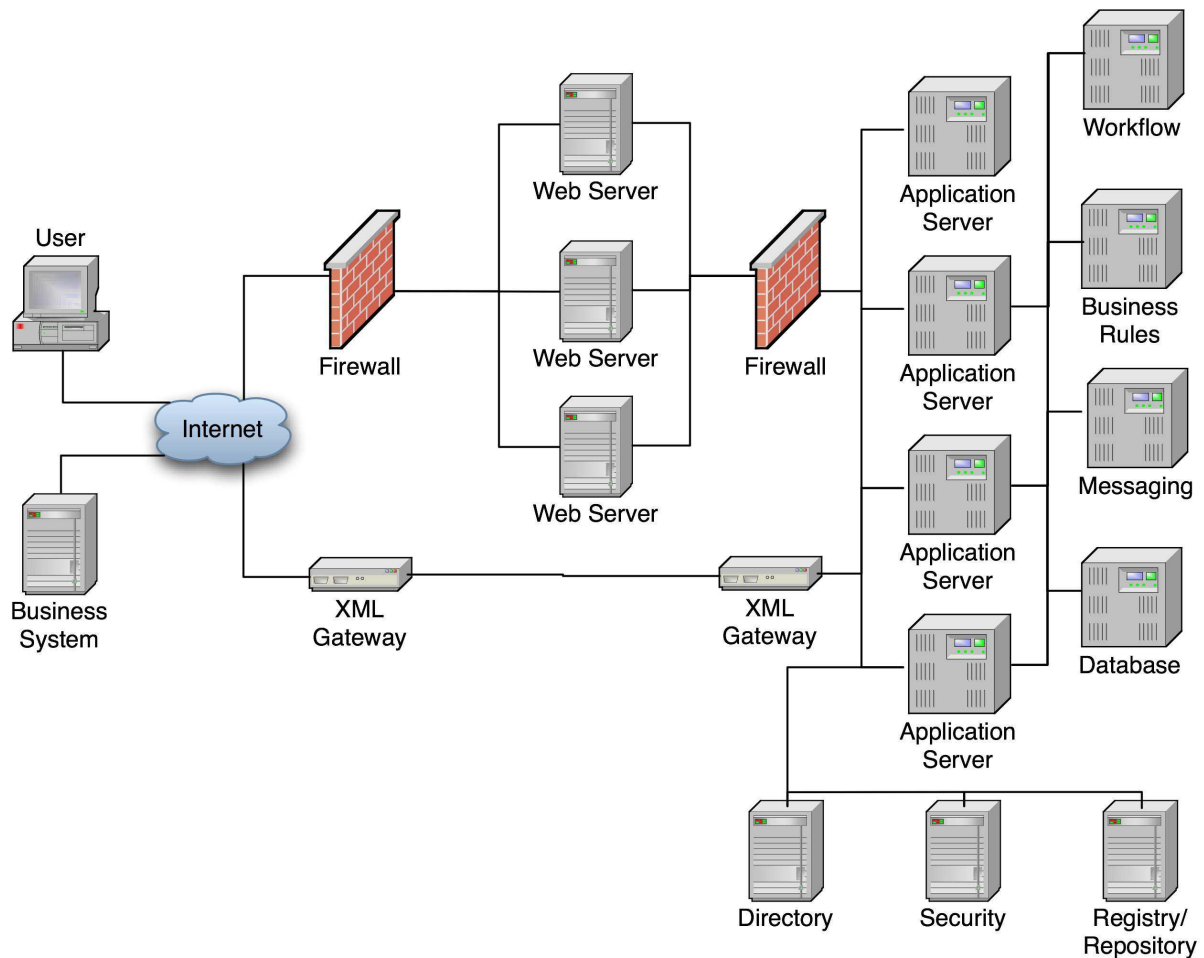
4.3.5.9 Email Server

The Email server is the infrastructure that provides a centralized location for the exchange of information through the sending and receiving of mail messages. In the J2EE platform JavaMail API is used to interact with the Messaging/Email servers. Some of the collaboration services like scheduling meeting, appointments and calendar services can also be accomplished using email/messaging servers. Some of the most popular email servers are the Sun ONE messaging server and the Lotus server.

4.3.6 Deployment View

The Deployment view demonstrates the network infrastructure view of the physical elements in the ITA. This infrastructure configuration aims to address some of the important considerations for systemic qualities (QoS), such as security, availability and scalability. The following section elaborates the major actors in the deployment view in perspective of the Systemic quality that they impact.

EOHHS CTO Organization	Version: 1.0
Secretariat Application Reference Manual	Date: February 02, 2011



- **Security**

Firewalls have been added to protect the trusted zone from the semi-trusted zone and the Internet.

Security Server will provide the services for authenticating and authorizing users accessing the application or service.

- **Availability**

Adding multiple instances of the Web Server, Application Server and Database server provides fail over capabilities and targets high availability.

- **Scalability**

Clustering the Web/Application Server enables load balancing and thus depicts horizontal scalability of the architecture.

- **Reliability**

EOHHS CTO Organization	Version: 1.0
Secretariat Application Reference Manual	Date: February 02, 2011

Standby and replication of the Database Server and Directory Server provides high reliability in the application architecture for the data integrity.

- **Maintainability**

Most of the infrastructure platforms that are used in this architecture are open standards compliant (J2EE) contributing to the ease of maintenance of the application.

- **Manageability**

The infrastructure platforms used in the architectural model (web/application servers) comes along with administration tools and utilities, which increases the manageability of applications deployed in them.

4.3.7 Technology Guidelines

In general, Open Source Frameworks should be used whenever possible. These frameworks often have a rich user community and the entry level costs are much lower. Support for these frameworks is often already purchased as part of other projects. The project team should consult with the Architecture team to determine if the open source framework being used already has a support contract or one must be purchased as part of the new project

4.3.7.1 Presentation Tier

4.3.7.1.1 Cascading Style Sheets (CSS)

Cascading Style Sheets (CSS) should be used to separate the document content from its presentation. The html in the web application needs to use well-defined classes. The class definition will be kept in a separate file (.css). The classes will conform and/or extend the Virtual gateway style sheet. The presentation attributes must not be defined in the html itself.

Version 2.1 will be used.

4.3.7.1.2 Supported Browsers

See ETRM at

[http://www.mass.gov/?pageID=afsubtopic&L=5&L0=Home&L1=Research+%26+Technology&L2=IT+Policies%2c+Standards+%26+Guidance&L3=Enterprise+Architecture&L4=Enterprise+Technical+Reference+Model+-+Service+Oriented+Architecture+\(ETRM+v.+5.0\)&sid=Eoaf](http://www.mass.gov/?pageID=afsubtopic&L=5&L0=Home&L1=Research+%26+Technology&L2=IT+Policies%2c+Standards+%26+Guidance&L3=Enterprise+Architecture&L4=Enterprise+Technical+Reference+Model+-+Service+Oriented+Architecture+(ETRM+v.+5.0)&sid=Eoaf)

Application features will degrade gracefully based upon features that are not supported by the browser. (See ETRM)

Because of HIPAA requirements, static content such as images, javascript, CSS or any HTML should not be cached on the client's browser. This can be accomplished by adding headers such as the no-cache header to the response.

If mobile support is required the development team should see the Enterprise Architecture team.

EOHHS CTO Organization	Version: 1.0
Secretariat Application Reference Manual	Date: February 02, 2011

4.3.7.1.3 Javascript

Javascript can be used within any part of the application with the following conditions:

- If Javascript is disabled, the application should still function per the requirements.
- No business validations can exist in Javascript only.
- The application must comply with section 508 requirements.
- The application must also meet WCAG version 1.0 guidelines

The guidelines for rich internet applications that meet accessibility requirements are being evaluated and will be published in future.

4.3.7.1.4 Ajax

AJAX based applications often do not comply with ADA requirements and should not be used. However, if the application requires AJAX, the project team should see the Enterprise Architecture team for review.

4.3.7.1.5 ActiveX Controls & Applets

ActiveX components or other browser features and applets should not be used.

4.3.7.1.6 Page Flow

Page flow should be done via Spring Web Flow. Spring Web Flow supports configurable flows that can be secured in a declarative manner. It also supports page composition using different mechanisms and Ajax rendering. It supports declarative binding and validation. It promotes modularity and reuse of flows.

Version 2.2.1 will be used.

More information is available at: <http://www.springsource.org/webflow>

4.3.7.1.7 MVC

Java Server Faces is a component based MVC presentation framework. The JSF life-cycle makes it easy for developers to plug in code that applies for that phase of the life-cycle. Custom components that provide common functionality that can be used by different applications should be developed and maintained.

Version 2.0 will be used. Myfaces should be used.

More information is available at: <http://myfaces.apache.org/>

4.3.7.1.8 Portal

For applications that require a portal server, it is recommended to use LifeRay. LifeRay is an open source portal platform that supports and integrates with other frameworks including Spring and Drools. LifeRay also provides tools for document management and collaboration.

-

EOHHS CTO Organization	Version: 1.0
Secretariat Application Reference Manual	Date: February 02, 2011

Any applications with social networking requirements should see the Architecture Team.

Version 6 must be used.

More information is available at: <http://www.liferay.com/>

4.3.7.2 Services Tier

Web services using SOAP will be used for any inter department or external interfaces. The web services will publish a Web Service Definition Language (WSDL) describing the web service, its operations, its elements, and its data types. WSDL will be maintained in source control so that versioning of the WSDL are available.

The web services should be written in such a way as to support multiple versions at the same time. This is done by having different bindings for the operations and using versioned namespaces for the XSDs.

The web services will use document literal - wrapped style for easy communication with .NET clients. Attachments should use Message Transmission Optimization Mechanism (MTOM).

Web service standards (WS-*) should be used whenever possible.

Many of these standards will be used by the enterprise services provided by the XML Gateway. This will include WS-Security for authenticating the calling application. WS-Addressing might be used by the XML-Gateway for routing purposes. The XML Gateway will manage all certificates as part of the enterprise certificate management and enterprise security solution.

More information on the versions adopted can be found at:

[http://www.mass.gov/?pageID=afterterminal&L=5&L0=Home&L1=Research+%26+Technology&L2=IT+Policies%2c+Standards+%26+Guidance&L3=Enterprise+Architecture&L4=Enterprise+Technical+Reference+Model+-+Service+Oriented+Architecture+\(ETRM+v.+5.0\)&sid=Eoaf&b=terminalcontent&f=itd_policies_standards_etrm5dot0_Application5dot0&csid=Eoaf](http://www.mass.gov/?pageID=afterterminal&L=5&L0=Home&L1=Research+%26+Technology&L2=IT+Policies%2c+Standards+%26+Guidance&L3=Enterprise+Architecture&L4=Enterprise+Technical+Reference+Model+-+Service+Oriented+Architecture+(ETRM+v.+5.0)&sid=Eoaf&b=terminalcontent&f=itd_policies_standards_etrm5dot0_Application5dot0&csid=Eoaf)

More information on web service standards is available at: <http://www.w3.org/2002/ws>

RESTful services should not be used for inter-agency communication.

4.3.7.2.1 Spring Framework

The Spring Framework is an open source light-weight Inversion of Control (IoC) container and can exist in a full JEE application server such as Weblogic Application Server. As an IoC container, Spring is responsible for managing object lifecycles: creating objects, calling initialization methods, and configuring objects by wiring them together.

Objects created by the container are also called Managed Objects or Beans. Typically, the container is configured by loading XML files containing Bean definitions which provide the

EOHHS CTO Organization	Version: 1.0
Secretariat Application Reference Manual	Date: February 02, 2011

information required to create the beans.

In many cases it's not necessary to use the container when using other parts of the Spring Framework, although using it will likely make an application easier to configure and customize. The Spring container provides a consistent mechanism to configure applications and integrates with almost all Java environments, from small-scale applications to large enterprise applications.

Spring supports annotations for transactional support. It also has hooks into a number of other frameworks while providing a loosely coupled architecture that does not bind you to any other particular framework.

The Spring framework supports Aspect Oriented Programming (AOP) by allowing the wiring of Aspects for cross-cutting concerns at runtime. This is in contrast to compile time aspects in frameworks like AspectJ which are more robust but also more difficult to use.

By using the Spring framework, annotations can be placed around the service methods to designate transaction boundaries. Spring can use the JTA Transaction manager of a full-blown JEE server or it can use its own local Transaction manager for transactions hitting a single data source.

Version 3.0.5 will be used.

More information is available at: <http://www.springsource.org/about>

4.3.7.2.2 Services Framework

Web Services should leverage Apache CXF. Apache CXF is an open source services framework. CXF helps you build and develop services using frontend programming APIs, like JAX-WS and JAX-RS. These services can speak a variety of protocols such as SOAP, XML/HTTP, RESTful HTTP, or CORBA and work over a variety of transports such as HTTP, JMS or JBI. Apache CXF integrates well with Spring and provides for either contract-first development. It also supports document-wrapped style services. Finally, the standards which are required by the Enterprise Services are also supported in the framework.

Version 2.3 will be used.

More information is available at: <http://cxf.apache.org/>

4.3.7.2.3 XML Data Binding

JAXB is the preferred binding framework.

Version 2.2.2 will be used.

More information is available at: <http://jaxb.java.net/>

4.3.7.2.4 Integration

Camel along with Service Mix provides a powerful open-source integration framework. The integration framework is based upon known Enterprise Integration Patterns (EIP) that integrates easily into Spring.

EOHHS CTO Organization	Version: 1.0
Secretariat Application Reference Manual	Date: February 02, 2011

Camel is a mediation framework that allows routing of the messages to desired endpoints. As a mediation framework, it is a critical part of a loosely coupled SOA architecture.

It provides a rich set of components that make it easy to receive messages from anywhere or to send the messages in a common way without tightly coupling your code to any other system.

Version 2.6 will be used

More information is available at: <http://camel.apache.org/>

4.3.7.2.5 Service Bus

A Service bus is used to provide a loose coupling between the application services and the messages. Only larger applications will use a service bus. When they are used, the Open Source Service Mix Service bus should be used.

Version 4.2.0 will be used

More information is available at: <http://servicemix.apache.org/home.html>

4.3.7.2.6 Persistence Framework

Enterprise Applications

Hibernate with JPA - Hibernate is a mature open source object-relational mapping (ORM) framework that can be used to map any relational database to an object domain model.

Consult with Architecture team if not using JPA spec

Hibernate framework incorporates many standard best practices such as prepared statements into the framework without the developer knowing about the details.

Best Practices

- Inheritance is supported
- surrogate keys should be used for all tables
- Composite keys should be avoided
- Use HQL whenever possible for maintenance reasons
- Use SQL for performance tuning

By using the HQL as the query language, the underlying database provider can change without any impact on the application code itself.

Queries should be externalized in their own hbm files. Poor performing queries should be written in native SQL. For those poorly performing queries, the queries should be tuned outside of the application and then added to the application once the SQL has been optimized. To determine the worst performing queries, use the AWR report or Oracle Tuning.

EOHHS CTO Organization	Version: 1.0
Secretariat Application Reference Manual	Date: February 02, 2011

Much of the Hibernate design was incorporated into the EJB3 specification and Hibernate can be used along with other JPA components as the EJB3 implementation. For EHS applications, Hibernate should be used as the JPA Entity Manager.

Version 3.6.0 will be used

More information is available at: <http://www.hibernate.org/>

Standard Applications

OpenJPA is an open-source persistence framework that implements the JPA specification - JSR - 317. It provides Object to Relational database mapping (ORM) using annotations.

Version 2.1 will be used.

More information is available at: <http://openjpa.apache.org/>

4.3.7.2.7 Workflow Engine

Applications should leverage a Business Process Engine when appropriate. jBPM is open source Business Process engine built and supported by JBoss. Process flows can be written in JPD or JBPM. Intelligent business processes can be defined with tasks for users to perform, activities done by the system, timers, scheduled jobs, sub-processes, etc.

The business process definition can be externalized from the code and allow for changes to the flow of information. The process definition is portable and can be deployed on multiple platforms.

jBPM can leverage Drools in order to make decisions about process flow.

Version 4.4 will be used.

More information is available at: <http://www.jboss.org/jbpm>

4.3.7.2.8 Job Schedulers

There are two kinds of schedulers: application job schedulers that run within the application process and system schedulers that outside of the application. For application scheduling, EHS applications should use the Quartz scheduler. For out of process scheduled jobs, Spring Batch should be used for scheduled jobs. Long running Quartz jobs should be created as stateful jobs. All Quartz jobs must handle exceptions gracefully and should include the ability to notify users (usually via e-mail) in the case of a failure. If there is more than one server running the jobs, Quartz should be configured in a cluster.

An alternative to Quartz is Spring Batch. The following from the Spring-batch document describes the Business Scenarios that Spring-Batch addresses.

- Commit batch process periodically

EOHHS CTO Organization	Version: 1.0
Secretariat Application Reference Manual	Date: February 02, 2011

- Concurrent batch processing: parallel processing of a job
- Staged, enterprise message-driven processing
- Massively parallel batch processing
- Manual or scheduled restart after failure
- Sequential processing of dependent steps (with extensions to workflow-driven batches)
- Partial processing: skip records (e.g. on rollback)
- Whole-batch transaction: for cases with a small batch size or existing stored procedures/scripts

Spring Batch version 2.1.3 will be used.

More information is available at: <http://static.springsource.org/spring-batch/>

Quartz version 1.8.5 will be used.

More information is available at: <http://www.quartz-scheduler.org/>

4.3.7.2.9 Logging and Logging services

Logging should be implemented using slf4j and logback.

Slf4j provides a façade for various logging frameworks. This allows the application to switch logging frameworks easily. It allows for tokenized strings which minimize the cost of string concatenation. It supports all standard logging levels of trace, info, debug, error, and fatal.

More information about sl4j is available at: <http://www.slf4j.org/>

More information about logback is available at: <http://logback.qos.ch/>

4.3.7.2.10 Caching

EhCache

Many applications require the use of a cache to improve performance. EHCACHE is an open source cache that is fully integrated into Hibernate. It can also be used with application servers to provide caching of other non-database objects.

EHCACHE also supports distributed caching where the cache spans multiple servers or where copies of the objects exist on more than one server. The latest version supports very large cache sizes that can scale beyond the size of the JVM. Finally, eviction policies can be set up so that objects can be removed from the cache according a policy that makes sense for the kind of object.

Version 1.7 will be used.

More information is available at: <http://ehcache.org/>

4.3.7.3 Database Tier

4.3.7.3.1 Database Server

EOHHS CTO Organization	Version: 1.0
Secretariat Application Reference Manual	Date: February 02, 2011

The standard for EHS applications is Oracle 10g Enterprise Version. Oracle database is available in 3 configurations. They are:

Standalone – This configuration is not scalable and failover capabilities are not available. This configuration is not recommended.

Oracle on linux cluster – This configuration provides failover capabilities in active passive mode. It is also the recommended configuration for applications deployed in virtual gateway.

Oracle Real Application Clusters (RAC) – This configuration provides scalability and failover features. This is the recommended configuration for very large mission critical applications.

All new applications must use Oracle as database. SQL Server is also supported. **Provide more guidance.**

4.3.7.3.2 Data Modeling Tools

The guidelines are being evaluated and will be published in future.

4.3.7.4 Security

4.3.7.4.1 Authentication

All applications must be integrated with the commonwealth enterprise solution for security called Access and Identity Management Services (AIMS). AIMS provides single sign-on capabilities, high level authorization and auditing at an enterprise level. Furthermore, AIMS provides a web service API.

Web Applications will either use the AIMS Authentication Server (Virtual Gateway) and AIMS agents or provide a JAAS module within Spring Security that uses the AIMS API calls for authentication.

For Web Services, WS-Security is used to initially authenticate the service caller. When the service has successfully authenticated, the caller will be given a token which will be used for subsequent calls. The XML Gateway will insert a SAML assertion into the web service call. The updated call will be sent to the backend web service provider. Using this paradigm, the web service provider will only need to manage the certificate of the XML Gateway.

4.3.7.4.2 Enterprise Authorization

The applications must use the Access Manager Agent to provide high level authorization into the application. The AIMS agent will enforce authorization to the application. Only users that are allowed to access a given application will be allowed into it. This level of authorization is provided via AIMS.

4.3.7.4.3 Application Authorization

Determining what a user can do once they have access to the application is application-level authorization. The application must provide its own authorization for such control.

4.3.7.4.4 Auditing

EOHHS CTO Organization	Version: 1.0
Secretariat Application Reference Manual	Date: February 02, 2011

Auditing for accessing the application is done by Enterprise Services. However, the application must also keep track of what actions a user performed within the application. In particular, all access to PHI must be captured and recorded. If PHI data is added, removed, or changed, the action must be tracked along with who and when the action occurred.

4.3.7.5 Reports

Reporting needs of an application are categorized into 2 types based on how they are generated and what is the data source. They are:

Reporting within the application – Reports are generated within the application using the OLTP database.

Reporting outside the application – Reports are generated outside the application using OLAP database.

4.3.7.5.1 Reporting within Application

For all JAVA based browser applications JASPER Reports is choice of EHS. JASPER Reports is a Java-based, open source report-generating tool that can deliver content onto the screen or printer, or into PDF, HTML, XLS, CSV and XML files.

More information is available at: <http://www.jaspersoft.com/jasperreports>

4.3.7.5.2 Reporting outside Application

Data warehouse reporting and analysis are done through a set of SAS tools. SAS provides a set of tools for aggregating and presenting the data gathered from the application. The data is then analyzed with a set of business intelligence tools and are reported.

COGNOS: The EHS chosen reporting tool, works with table relationship designs specifically found in data warehouses and similar sources. A COGNOS service is available through Virtual Gateway which allows authorized VG users run any COGNOS report. COGNOS offers two methods to generate reports:

Report Studio: It is a reporting tool that allows designated report developers to develop complex reports. It can also be used for quick ad-hoc queries and analysis.

Query Studio: A simpler tool than Report Studio, it can be used for ad-hoc queries and simple queries and learned more easily.

4.4 Requirements

The process and guidelines are being evaluated and will be published in future.

4.5 Development

4.5.1 Integrated Development Environment

Eclipse is an IDE that provides tools for developing Java EE and web applications.

Version 3.6.2 – Helios will be used.

More information is available at: <http://www.eclipse.org/>

EOHHS CTO Organization	Version: 1.0
Secretariat Application Reference Manual	Date: February 02, 2011

4.5.2 Project Structure

Project structure should follow Maven standards. Maven defines a standard project structure and allows common components to be shared across applications. Maven uses a repository to hold the modules used in the application. Modules developed for the Commonwealth can be shared by added then to a common repository.

Properly built Maven projects also allow modules within the same project. This provides a way for the code to be broken down into functional units where each module deals with a specific set of functions. Other modules may have dependencies to the modules but they do not need to include all the source code in each build. These modules allow developers to focus on the code in their functional area without needing to build out all the other modules.

Maven is also used as a build tool and can be used with other build tools such as Ant and it works well with continuous integration environments like Hudson.

Maven also supports the existence of multiple environments easily. Each environment can have its own set of resources and configuration. This structure allows environment specific information to remain isolated.

While working in Eclipse, install the m2eclipse plug-in for Maven builds and deployments from Eclipse.

src/main/java	Application/Library sources
src/main/resources	Application/Library resources
src/main/filters	Resource filter files
src/main/assembly	Assembly descriptors
src/main/config	Configuration files
src/main/webapp	Web application sources
src/test/java	Test sources
src/test/resources	Test resources
src/test/filters	Test resource filter files
src/site	Site
LICENSE.txt	Project's license
NOTICE.txt	Notices and attributions required by libraries that the project depends on
README.txt	Project's readme

More information is available at: <http://maven.apache.org/guides/index.html>

EOHHS CTO Organization	Version: 1.0
Secretariat Application Reference Manual	Date: February 02, 2011

4.5.3 Build Tool

Ant is a widely used tool for building Java based applications. It can be used either alone or within Maven to build and deploy the application. Additionally, Ant can be used within Hudson integrated build environment and within the Eclipse IDE.

Version 1.8.2 will be used.

More information is available at: <http://ant.apache.org/>

4.5.4 Continuous Integration

Jenkins provides an easy-to-use continuous integration system, making it easier for developers to integrate changes to the project, and making it easier for users to obtain a fresh build. The automated, continuous build increases the productivity.

Version 1.406 will be used.

More information is available at: <http://jenkins-ci.org/>

4.5.5 Code Quality

Check Style

Check Style is a development tool to help programmers write Java code that adheres to a coding standard. By default it supports the Sun Code Conventions but it can be configured to support almost any coding standard.

Version 5.3 will be used.

More information is available at: <http://checkstyle.sourceforge.net/>

Find Bugs

Findbugs is a tool used to identify potential bugs within code. It inspects the bytecode for bug patterns.

Version 1.3.9 will be used.

More information is available at: <http://findbugs.sourceforge.net/>

PMD

PMD is a static analyzer that scans java source for problems. Capabilities include scanning for empty block statements, dead code, suboptimal code, overcomplicated expressions and duplicate code.

Version 4.2.5 will be used.

More information is available at: <http://pmd.sourceforge.net/>

EOHHS CTO Organization	Version: 1.0
Secretariat Application Reference Manual	Date: February 02, 2011

4.5.6 XML Editor

XML Spy is an XML Editor and validator. It can be used to create XML Schemas and WSDLs. All WSDLs and their XML schemas need to be validated with XML Spy before being published to the XML Gateway.

Please note XML Spy is a licensed product that requires procurement before using.

Version 2011 will be used.

Eclipse Plugin

The Eclipse XML editor can be used for initial development of the XML Schemas and the WSDLs. However, XML Spy has a better track record for validation of the schemas and the ability to run in the DataPower XI-50.

4.6 Testing

All the applications must follow the process adopted by the Software Quality Assurance team. The following table provides the details of this process:

EOHHS SQA TEST PROGRESSION MATRIX v 1.0							
	UNIT Testing	Integration Testing	Functional Testing	Technical Testing	User Acceptance	Production	
	Testing the smallest executable units	Testing the units integrated into the build perform correctly	Independent validation of feature functionality across entire system (End to End) Screen Reader - JAWS	Performance Stress Vulnerability Accessibility	Subset of test artifacts executed by User Community against real world business scenarios	Deployment Verification Navigation Testing Controlled Production Input	
Environment	Desktop	EOHHS DEV	EOHHS ST	EOHHS QA ENV	EOHHS QA ENV	EOHHS PRD	
Defect Tracking Tool	DEV Choice	ClearQuest	ClearQuest	ClearQuest	ClearQuest	ClearQuest	
Test Artifact Management	Selenium	EOHHS Test Manager	EOHHS Test Manager	Loadrunner, APP SCAN, Policy Tester	EOHHS Test Manager	Selenium/ Manual	
Change Management	MF	MF	MF	MF	MF	MF	
TASKS	RESPONSIBILITY						
Create Test Plan		Development Team	EOHHS SOA	EOHHS SOA	SOA/ End Users	EOHHS SOA	
Develop Test Cases	Developer	Development Team	EOHHS SOA	EOHHS Tech SOA	EOHHS End Users	EOHHS SOA	
Test Data Creation	Developer	Development Team	Dev Team/ SOA	Development Team	Dev Team / End Users	End Users	
Execute Test Cases	Developer	Development Team	EOHHS SOA	EOHHS Tech SOA	EOHHS End Users	EOHHS SOA	
Support Testing	Development Team	Development Team	Development Team	Development Team	DEV / SOA Teams	Development Team	
Manage Defect Fixes	Development Team	Development Team	PM/SQA/DEV/BIZ	EOHHS ARCH TEAM	PM/SQA/DEV/BIZ	EOHHS SOA	
Produce Test Report	Dev Lead	Development Team	EOHHS SOA	EOHHS SOA	SOA/ End Users	RM	
Review Test Results	Dev Lead	EOHHS SOA	PMO/SQA/ARCH/RM	EOHHS ARCH TEAM	PMO/SQA/ARCH/RM	EOHHS OPS	
Milestone Scheduling	Dev Lead	PMO	PMO	PMO	PMO	RMPMO	
Code Promotions		Development Team	RM	RM	RM	RM	
Create Test Plan	Document Test Plan in EOHHS Standard MASTER TEST PLAN format						
Develop Test Cases	Document test steps, verification points and expected results in Test Manager software. Test Cases should be to the level of detail where a person with no prior knowledge of the system can execute. All Test Cases should map back to Use Cases/ Requirements via test coverage matrix						
Execute Test Cases	Management of test data needed to execute all test case scenarios. Includes: Data Migration approach, PHI scrubbing (where applicable) for all environments.						
Test Data Creation	Physical execution of test cases in Test Manager. Tested by & Pass / Fail actual vs. expected results. All test anomalies recorded in Clearquest						
Support Testing	Answer questions / provide technical and subject matter expertise during test cycle execution						
Manage Defect Fixes	Triage and target for next logical build for resolution. Assign to DEV staff for fix. Update status and resolution notes in Clearquest. Goal to meet QUALCON entry and exit criteria						
Produce Test Report	Provide Detailed planned vs. actual test execution progress reports. Include final test summary per test cycle with quantifiable coverage metrics and code readiness assessment.						
Review Test Results	Inspection of Test Artifacts and Test Summaries. Impact assessment of test anomalies (failures). Acceptance/ Rejection of potential show stopper issues.						
Milestone Scheduling	Scheduling of milestone reviews, code drops and go no go meetings based on readiness assessment of artifacts posted to MassForge						
Code Promotions	Version Control. Check in Check out of code base assets. Migration of code from environment to environment utilizing step by step Env. Specific documented task lists. Managing coordination of DBA, Security and Integration Services						

EOHHS CTO Organization	Version: 1.0
Secretariat Application Reference Manual	Date: February 02, 2011

4.6.1 Unit Testing

Java

JUnit is a simple, open source framework to write and run repeatable tests. It will be used to create the unit tests for java code.

Version 4.8.2 will be used.

More information is available at: <http://junit.org/>

Web Services

SoapUI is a tool for functional testing, mainly of Web Services like SOAP based Web Services and REST Web Services, but also HTTP based services and JMS Services. SoapUI has a commercial offering called SoapUI Pro, with extra functionality.

Version 3.6.1 will be used.

More information is available at: <http://www.soapui.org/>

File Transfers

CURL is used to test large xml files against the XML Gateway. SOAPUI can be used for smaller XML payloads.

Version 2.2.1 will be used.

More information is available at: <http://curl.haxx.se/>

HIPAA Transactions

OnlyConnect FirstPass is a windows desktop application that validates X12 EDI data for adherence to X12 syntax and other requirements as specified in the 5010 X12 Type 3 Technical Reports (TR3) adopted under HIPAA.

Web Page Inspection and Debugging

Firebug allows editing, debugging, and monitoring of CSS, HTML, and JavaScript live in any web page. It can be used to monitor request activity, visualize CSS metrics, information about errors in JavaScript, CSS and XML. Includes DOM explorer; execute JavaScript on the fly.

Version 1.7 will be used.

More information is available at: <http://getfirebug.com/>

TCP/IP Monitor

Use the TCP/IP monitor built into Eclipse or SOAPUI to see the payload and headers exchanged between the Web service client and the server. It can also be used to see browser based traffic between the browser and the server.

EOHHS CTO Organization	Version: 1.0
Secretariat Application Reference Manual	Date: February 02, 2011

HTTP Debugging

Fiddler is a HTTP debugging proxy which logs all HTTP traffic between your computer and the Internet. Fiddler allows you to inspect all HTTP Traffic, set breakpoints, and "fiddle" with incoming or outgoing data. Fiddler includes a powerful event-based scripting subsystem, and can be extended using any .NET language. It can debug traffic from virtually any application and available only for Windows platforms.

Version 2 will be used.

More information is available at: <http://www.fiddler2.com/fiddler2/>

4.6.2 Functional Testing

Test/Code Coverage Analysis

Cobertura is tool to identify which parts of a Java program are lacking test coverage and calculate % code coverage. It also reports McCabe cyclomatic complexity of the code.

Version 1.9.4 will be used.

More information is available at: <http://cobertura.sourceforge.net/>

Acceptance Tests

FitNesse is a lightweight, open-source framework that makes it easy for software teams to collaboratively define Acceptance Tests. Web pages containing simple tables of data inputs and expected outputs are used to run tests and see the results. The tables are expressed in the form of a wiki; the Fitness wiki enables easy creation and editing of the Wiki/Fitness pages. FitNesse is a web server.

Release 20110104 will be used.

More information is available at: <http://fitnesse.org/>

4.6.3 Regression Testing

Web application

Canoo WebTest is a tool that tests the expected html against the result html. Since it compares expected html against result html, it can be used for regression tests. Reporting includes high level summary reports to details of each test.

Features

- Reporting
- Test scripts in Groovy or in XML
- Test Suites
- Integration with continuous integration tools like Hudson.

EOHHS CTO Organization	Version: 1.0
Secretariat Application Reference Manual	Date: February 02, 2011

Version 3.0 will be used.

More information is available at: <http://webtest.canoo.com/>

Selenium is a suite of tools that allow web testing. Its firefox add-in based IDE allows the developer to record clicks, typing, and other user interactions into a "script" and replay it back for developer testing. The developers can also use Selenium to verify that the UI still functions to the specifications.

Version 1.0.10 will be used.

More information is available at: <http://seleniumhq.org/>

Database

DBUnit is a tool used for testing that puts a database in a known state between test runs. This allows tests to be repeated with a known set of data in the database.

Features

- export your database data to XML datasets
- import your database data from XML datasets
- work with very large datasets when used in streaming mode
- verify that your database data match an expected set of values as part of a test

Version 2.4.8 will be used.

More information is available at: <http://www.dbunit.org/>

Web Services

As mentioned above in the Unit Test section, JUnit can be used for regression tests to make sure that the service layer code meets the requirements as enhancements and bug fixes are made.

SOAPUI is used to verify the web services between and as part of releases. See the section on SOAPUI earlier in the document.

4.6.4 Performance Testing

Load Runner will be used for load/stress testing tool for web and other applications. It supports a wide variety of application environments, platforms, and databases. It has a large suite of network/app/server monitors to enable performance measurement of each tier/server/component and tracing of bottlenecks.

Version 11.2 will be used.

EOHHS CTO Organization	Version: 1.0
Secretariat Application Reference Manual	Date: February 02, 2011

4.6.5 Accessibility Testing

ADA Compliance

IBM Policy Tester will be used for compliance testing to assess quality, privacy, and accessibility compliance issues across corporate web properties.

Version 5.5 will be used.

Screen Reader

The web content provided by either the web site or web application must work with an accessibility solution that reads information on your screen using synthesized speech. Freedom Scientific's JAWS will be used.

Version 12 will be used.

Color Contrast Testing

Juicy Studio is the tool that will be used for color contrast checks.

Version 1.6 will be used

Web Accessibility Toolbar is IE plug-in that aids in manual examination of web pages for a variety of aspects of accessibility. It consists of a range of functions that:

- identify components of a web page
- facilitate the use of 3rd party online applications
- simulate user experiences
- provide links to references and additional resources
- color contrast checks.

Version 2.0 will be used.

4.6.6 Vulnerability Testing

IBM rational AppScan will be used to test web application security. Testing includes both

- Dynamic analysis to test for all common web application vulnerabilities
- Static analysis of JavaScript to identify client-side vulnerabilities

Version 8.0.x will be used.

4.6.7 Defect Tracking

The guidelines are being evaluated and will be published in future.

ClearQuest will be used for defect tracking.

EOHHS CTO Organization	Version: 1.0
Secretariat Application Reference Manual	Date: February 02, 2011

4.6.8 Performance Monitoring

4.6.8.1 *Application Monitoring*

4.6.8.1.1 Wily

Wily is an Application Performance Management tool made by Computer Associates (CA). It provides a view to the performance of the application from the browser to the database. It can inspect and provide metrics at a component level for both Java and .NET applications. There are also power packs for application servers such as Oracle Weblogic and IBM WebSphere.

Version 8.2.3 will be used.

4.6.8.1.2 JRockit Mission Control & Runtime Analyzer

Features

- Memory Leak Detector
- Eclipse Plug-in
- Identify hotspots
- Exception Counters
- Profiling information
- Stack traces for selected threads

The JRockit Runtime Analyzer (JRA) can be used to check the overall health of the JVM. It is important to note that JRA only works with the Oracle JRockit JVM. JRA supports real-time monitoring and captures over a specified period of time. Captures can be analyzed can then be used to tune the JVM so that it runs efficiently.

Some of the key areas to look at from the JRA recording are

- thread local area (TLA) size
- size of the largest object allocated
- size of the heap
- frequency of the garbage collections
- locks

The JRockit documentation provides some good guidelines and should be consulted during the tuning phase of application delivery. Also check with the Architecture team for some tuning recommendations

4.6.8.1.3 HP Open View

HP Open View should be used for monitoring memory, CPU, file system usage. It can be accessed using the following URL:

EOHHS CTO Organization	Version: 1.0
Secretariat Application Reference Manual	Date: February 02, 2011

<http://ovopvm1.itd.state.ma.us:8081/OVPM/Analyzer>

ITD is phasing this out and replacing with Tivoli monitoring. The guidelines are being evaluated and will be published in future.

4.6.8.2 DB Monitoring

4.6.8.2.1 Oracle Tools

The standard application used for tuning the database is the Oracle set of tools including

- Automatic Workload Repository (AWR) Reports
- Oracle Performance Pack
- Oracle Diagnostic Pack
- Oracle Automatic Database Diagnostic Monitor.

4.6.8.2.2 Toad

Toad is a tool-suite provided by Quest. It has versions for developers, DBAs, and analysts. The DBA tools can be used to tune both the Oracle DBMS and it can also identify SQL statements that are performing poorly. Toad can take a poorly performing query and test the various ways in which the same statement could be written and come up with the best performing permutation of the query that gives the same results.

Version 10.6.1 will be used.

4.6.8.2.3 ION

(Workload Interface Statistics Engine) - Ion is the only tool outside OEM that will plot time-series data from AWR and ASH. It allows you to view Oracle trends by day-of-the-week and hour-of-the-day. The Ion EE also allows the same interface to Oracle8i and Oracle

Features

- Fast drill-down into table access details
- Built-in intelligence finds problems fast
- Spot trends with time-series regression
- See all components of Oracle response time
- Easily spots hidden resource bottlenecks
- Predict problems before they cripple your database.
- Supports Oracle 9i through 11g.

4.7 Environment

4.7.1 Application Categories

Each application must be reviewed by the architecture team to determine its classification as either an Enterprise application or a standard application.

EOHHS CTO Organization	Version: 1.0
Secretariat Application Reference Manual	Date: February 02, 2011

Enterprise Applications

Enterprise applications are mission critical applications with a large user base and high performance needs. These applications will be deployed on Weblogic application servers.

Standard Applications

Standard applications are non-mission critical applications, with a small user base. They do not have highly active development cycles. These applications will be deployed on JBoss application servers.

4.7.2 Application Environments

There will be 5 environments in which the application will be deployed during its life cycle. They are:

- Development – This is used by the development team during construction phase. All development activities including unit testing will be executed in this environment. The application test results will be reviewed and upon approval will be ready for promotion to System Test environment.
- System Test – This is used by the development team for integration testing. The application under development will be deployed and integrated with other systems. Testing focused on the integration aspects will be performed in this environment. The test results will be reviewed and upon approval will be ready for promotion to QA environment.
- QA – This is used by QA team for load testing, capacity planning, accessibility, vulnerability testing. The application in its integrated state will be subject to performance tests, to identify any performance issues. Once tuned for performance service level agreements, the application will be subjected to load. Once tuned for performance, the operating capacity of the application is identified. The application is also subject to different tests to identify accessibility and vulnerability issues. Once the results are review and upon approval will be ready for promotion to production.
- Production – This is the environment where the application is available for business users.
- Training – This is the environment where the application is made available for training business users.

4.7.3 Virtual Gateway

It provides the necessary infrastructure for deploying applications that adheres to enterprise security policies. It consists of the following:

- A cluster of Weblogic application servers
- A cluster of JBoss application servers
- A cluster of Apache web servers in a DMZ with mod_security
- An XML gateway
- A cluster of oracle database servers supporting high availability in active/passive configuration

All applications based on JEE will deployed in virtual gateway.

The following links provide details of virtual gateway infrastructure in different environments:

Development & System Test: <https://massforge.state.ma.us/sf/go/doc23273?nav=1>

EOHHS CTO Organization	Version: 1.0
Secretariat Application Reference Manual	Date: February 02, 2011

QA: <https://massforge.state.ma.us/sf/go/doc23271?nav=1>

Production: <https://massforge.state.ma.us/sf/go/doc22726?nav=1>

Training: <https://massforge.state.ma.us/sf/go/doc23272?nav=1>

4.7.4 Virtual Datacenter

It provides VMWare based virtualized environment for deploying applications that adheres to enterprise security policies. It consists of the following:

- Infrastructure divided into 4 different security zones (Public Internet, Production Internal, Non-Production, ILO).
- Follows a charge back model for usage of the infrastructure

All applications that have infrastructure requirements, licensing model that cannot be met by the VG environment will be deployed in virtual data center. For example: .Net applications, Fax Servers.

4.8 Release Management

The guidelines are being evaluated and will be published in future.

4.9 Configuration Management

Even though there are multiple source management models available, EHS will be follow a centralized model where all the revision control functions take place on a shared server. All EHS applications should use Subversion as the version control software. This provides the following advantages:

- Atomic Commits
- Optimistic Locking
- Version Merging
- Base lining, Labeling and Tagging
- Distributed Version Control
- Integration with Multiple IDEs
- Integration with MassForge

4.10 Change/Risk/Issue Management

The guidelines are being evaluated and will be published in future.

EOHHS CTO Organization	Version: 1.0
Secretariat Application Reference Manual	Date: February 02, 2011

5. Helpdesk Services

The processes and guidelines are being evaluated and will be published in future.